

AMENDMENT TO THE CLAIMS

1. (Currently Amended) A method of sharing ~~at least one~~ a resource amongst a plurality of applications issuing requests in different request classes, comprising the steps of:

i) dynamically assigning a priority to each of a plurality of request queues associated with respective ones of said request classes, such assigning being in accordance with a moving average resource allocation to each of said respective request classes and the priority assigned to a respective request queue being a function of the moving average resource allocation to the associated request class; inversely proportional to a moving average resource allocation to each of said respective request classes;

ii) receiving and queuing said requests from said applications in said plurality of request queues in accordance with said respective request classes;

iii) allocating said ~~at least one~~ resource to one of said applications whose request has been queued longest in a highest priority one of said ~~plurality of request queues; and~~ and in response to said one of said applications relinquishing said resource; then

iv) in response to said one of said applications relinquishing said resource, repeating steps i) to iii).

2. (Currently Amended) The method of claim 1, wherein said step of dynamically assigning said priority to each of said plurality of request queues is implemented as follows:

$$\begin{aligned}
 p_i &= 1.0 && \text{if } u_i \leq \min_i \\
 p_i &= 1.0 - (u_i - \min_i) / (\max_i - \min_i) && \text{if } \min_i < u_i \leq \max_i \\
 p_i &= 0.0 && \text{if } u_i \geq \max_i
 \end{aligned}$$

where p_i is the priority assigned to the i^{th} one of said queues associate with the i^{th} one of said request classes, and u_i is the moving average allocated to said i^{th} one of said request classes and \min_i and \max_i are respectively the minimum and maximum allocations to said i^{th} one of said request classes.

3. (Currently Amended) The method of claim 2, further including the step of updating said moving average of each of said request classes immediately upon allocating said resource, ~~in the event of multiple resources.~~

4. (Currently Amended) The method of claim 1, wherein the priority assigned to each of said plurality of request queues in accordance with said moving average resource allocation conforms to a predetermined linear function, ~~shape of priority function.~~

5-7. (Cancelled)

8. (New) The method of claim 1, wherein the priority assigned to each of said plurality of request queues in accordance with said moving average resource allocation conforms to a predetermined exponential function.

9. (New) The method of claim 1, wherein the priority assigned to each of said plurality of request queues in accordance with said moving average resource allocation conforms to a predetermined step function with steps of varying size.